Computable counter-examples to the Brouwer fixed-point theorem

Petrus H. Potgieter

Department of Decision Sciences, University of South Africa (Pretoria), PO Box 392,
Unisarand, 0003, Republic of South Africa, php@member.ams.org,
potgiph@unisa.ac.za, www.potgieter.org

Abstract. This paper is an overview of results that show the Brouwer fixed-point theorem (BFPT) to be essentially non-constructive and non-computable. The main results, the counter-examples of Orevkov and Baigger, imply that there is no procedure for finding the fixed point in general by giving an example of a computable function which does not fix any computable point. Research in reverse mathematics has shown the BFPT to be equivalent to the weak König lemma in RCA_0 (the system of recursive comprehension) and this result is illustrated by relating the weak König lemma directly to the Baigger example.

Keywords: Computable analysis, Brouwer fixed-point theorem, weak König lemma

1 Introduction

We consider the Brouwer fixed-point theorem (BFPT) in the following form, where the standard unit interval is denoted by $I = [0, 1]$.

Theorem 1 (Brouwer). *Any continuous function $f : I^2 \to I^2$ has a fixed point, i.e. there exists an $x \in I^2$ such that $f(x) = x$.*

A computable real number is a number for which a Turing machine exists that, on input n, produces a rational approximation with error no more than 2^{-n}. A computable point is a point all the coordinates of which are computable reals. The notation

$\mathbb{N}_0$ for the non-negative natural numbers;
$\mathbb{R}_c$ for the set of computable reals;
I_c for $I \cap \mathbb{R}_c$; and
δX for the boundary of a set X, being $\overline{X} \cap \overline{X^c}$

is also used. The two examples discussed use distinct definitions of a computable function of real variables.

Russian school In the Russian school of Markov and others, a computable function maps computable reals to computable reals by a single algorithm for the function that translates an algorithm approximating the argument to an algorithm approximating the value of the functions. It need not be possible to extend a function that is computable in the Russian school to a continuous function on all of the reals. These functions are often called *Markov-computable.*

Polish school In the Polish school of Lacombe, Grzegorczyk, Pour-El and Richards, and others, a function is computable on a region if it maps every every computable sequence of reals to a computable sequence of reals and it has a computable uniform modulus of continuity on the region [1].

2 Orevkov's example for the Russian school

One can construct a Markov-computable function f through a computable mapping of descriptions of computable points $x \in I_c^2$ to descriptions of $f(x) \in I_c^2$, such that

$$f(x) \neq x \qquad \forall x \in I_c^2.$$

That is, no computable point is a fixed point for f. Unfortunately the f which is constructed in this way, cannot be extended to a continuous function on I^2. This is the construction of [2], another instance of which can be found in [3].

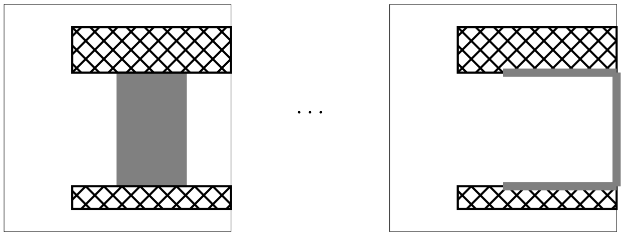

Fig. 1. Basic contraction in the Orevkov counter-example

Lemma 1. *Suppose A_k is a sequence of rectangles in I^2 with computable vertices, disjoint interiors, and such that*

(i) $\emptyset \neq \delta A_j \setminus \bigcup_{i<j} A_i$ *for all* j*;*

(ii) for each j *there exists* $n > j$ *such that* $\delta A_j \subset \left(\bigcup_{i \leq n} A_i\right)^{\circ}$*; and*

(iii) $I_c^2 \subseteq \bigcup_{i \geq 1} A_i$

then there exists a Markov computable g*, mapping* I_c^2 *to* δI_c^2 *and fixing* δI_c^2*.*

The conditions ensure that

(i) rectangles A_j, when added, have some part of their boundary in $I^2 \setminus \bigcup_{i<j} A_i$;
(ii) each A_j is eventually closed off by new rectangles on all sides;
(iii) all computable points lie in $\bigcup_{i\geq 1} A_i$.

The function f is obtained by composing g with a 90° rotation. It therefore remains only to prove the lemma and the existence of a sequence of rectangles which is as required. Suppose that g has been defined on $\bigcup_{i<j} A_i$. For

$\emptyset = \delta A_j \cap \bigcup_{i<j} A_i$: let g on A_j consist of the simplest possible mapping to δI^2, that fixes δI^2;

$\emptyset \neq \delta A_j \cap \bigcup_{i<j} A_i$: we extend g to A_j by using (i)—if g has already been defined on the crosshatched set in Figure 1 then the definition can be extended to the solid gray set A_j by composing a contraction of the solid gray set in Figure 1 to

$$\left(\delta A_j \cap \bigcup_{i<j} A_i\right) \cup \left(\delta A_j \cap \delta I^2\right)$$

with the function g as it has already been defined on the crosshatched set. This is always possible because, by construction of the A_i, our A_j will always have at least *two* sides non-contiguous with $\bigcup_{i<j} A_i$, at least one of which will not coincide with δI^2.

So far only condition (i) has been used. Conditions (ii) and (iii) are necessary for showing that g is Markov-computable on I^2. Let a description of any $x \in I^2_c$ be given. We can find a description of $g(x)$ in the following way.

- Simultaneously, compute approximations of x using the given description and construct g on $\bigcup_{i\leq n} A_i$ for $n = 1, 2, \ldots$.
- Together, (ii) and (iii) imply that for some n we will be able to verify that

$$x \in \left(\bigcup_{i\leq n} A_i\right)^{\circ}$$

  where the interior is with respect to the subset topology on I^2, of course.
- When such an n has been identified, we already know the definition of g for $\bigcup_{i\leq n} A_i$ as well as the modulus of continuity of g on the same set. This is now used to describe $g(x)$.

It remains to be shown that a suitable sequence of rectangles $(A_n)_{n\geq 1}$ exists. This follows from the next fact, assumed without proof for now[1].

Lemma 2 (see [4], for example). *There exist computable sequences of rational numbers (a_n) and (b_n) in the interval $I = [0, 1]$ such that the intervals $J_n = [a_n, b_n]$ have the following properties.*

[1] Later we shall deduce the fact from the existence of a Kleene tree.

(i) *If $n \neq m$ then $|J_n \cap J_m] \leq 1$.*
(ii) *If $a_n \neq 0$ then $a_n \in \{b_0, b_1, \ldots\}$ and if $b_n \neq 1$ then $b_n \in \{a_0, a_1, \ldots\}$.*
(iii) *$I_c \subsetneq \bigcup_n J_n$, i.e. the J_n cover the computable reals in $I = [0, 1]$.*

Now, let $(A_n)_{n \geq 1}$ be any computable enumeration of the $J_k \times J_\ell$. This completes the proof of the lemma, and the example.

3 Baigger's example for the Polish school

Let a be any non-computable point in I^2. Consider the function f which moves each point half-way to a,

$$f(x) = x + \frac{1}{2}(a - x)$$

and has a single fixed point, namely a itself. The function f is continuous and defined on all of I^2 and has no computable fixed point. Nevertheless, this is not really interesting since

- the fixed point a has no reasonable description—since it is itself not computable; and therefore
- the function f has no reasonable description—it is not computable in any sense.

One would like to see a function which is computable, defined (and therefore continuous) on all of I^2 and yet avoids fixing any of the computable points I^2_c. The following example, having appeared in [5] and in [3], modifies the construction of Orevkov to produce a computable f defined on all of I^2 having no computable fixed point. One uses the intervals $J_n = [a_n, b_n]$ of Orevkov's example and sets

$$C_n = \bigcup_{k,\ell \leq n} J_k \times J_\ell$$

after which one defines f progressively, using the sets C_n. The points

$$t_n = (v_n, v_n)$$

where

$$v_n = \min_{x \in I} \{x \mid (x, x) \notin C_n\}$$

are used as "target point" at each stage of the construction, as in Figure 2. Note that

$$v = \lim_{n \to \infty} v_n$$

is not a computable number and (v, v) will be one of the fixed points of f.

Definition 1. *For any $W \subseteq I^2$ we define*

$$W^{\blacksquare\varepsilon} = \left\{x \in W \;\middle|\; d\left(x, \delta W \setminus \delta I^2\right) \geq \varepsilon\right\}$$

and

$$W^{\square\varepsilon} = \left\{x \in W \;\middle|\; d\left(x, \delta W \setminus \delta I^2\right) = \varepsilon\right\}.$$

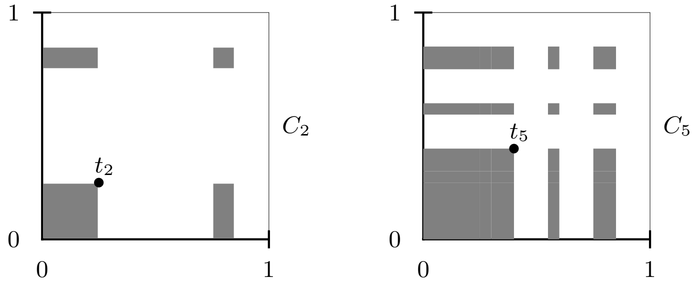

Fig. 2. The “target points” t_n

One can define f_n such that

1. f_n moves every point in the *interior* of $C_n^{\blacksquare 2^{-n}}$ but is the identity outside the set, and is computable;
2. f_{n+1} agrees with f_n on $C_n^{\blacksquare 2^{-n}\cdot\frac{3}{2}}$ and therefore
3. $f = \lim_{n\to\infty} f_n$ is computable.

Every computable point eventually lies in some

$$C_n^{\blacksquare 2^{-n}\cdot\frac{3}{2}} \quad \subset \quad \left(C_n^{\blacksquare 2^{-n}}\right)^{\circ}$$

and is therefore moved by f. Clearly $f(I^2) \subseteq I^2$ and f will be as required. In fact, f has no fixed point in

$$\bigcup_n C_n = \bigcup_{k,\ell\geq 1} J_k \times J_\ell .$$

Also, f has no isolated fixed point—its fixed points all occur on horizontal and vertical lines spanning the height and breadth of the unit square. Further details of the construction appear in Appendix A. The construction cannot be applied in the one-dimensional case because it is impossible to effect a change of direction by continuous rotation.

4 BFPT and the König lemma

In reverse mathematics it is known that in RCA_0, the system of recursive comprehension and Σ^0_1 -induction, the weak König lemma, WKL_0, is equivalent to the Brouwer FPT [6].

Lemma 3 (WKL$_0$, Kőnig). *Every infinite binary tree has an infinite branch.*

The König lemma does not have a direct computable counterpart.

Theorem 2 (Kleene [7]). *There exists an infinite binary tree, all the computable paths of which are finite.*

The relation of the Kleene tree to the Baigger counterexample is reviewed in this section. The discussion is informal and attempts only to give the essential ideas that have been revealed by the approach of reverse mathematics. In RCA_0, the weak König lemma WKL_0 has been shown to be equivalent to a number of other results in elementary analysis, such as the fact that any continuous function on a compact interval is also uniformly continuous [8]. WKL_0 and RCA_0 can, furthermore, be used to prove Gödel's incompleteness theorem for a countable language [9].

4.1 From Baigger f to Kleene tree

Let f be a computable function, as in the Baigger example, mapping I^2 to itself—with no computable fixed point. The following auxiliary result will be used to construct the Kleene tree.

Lemma 4. *Let a computable $g : I^2 \to [0,1]$ be given. Then there exists a Turing-computable $h : \mathbb{N}_0^9 \to \mathbb{N}_0^2$ such that for any $(n_1, n_2, \ldots, n_8, k)$ with*

$$0 \leq \frac{n_1}{n_2} \leq \frac{n_3}{n_4} \leq 1 \quad \textit{and} \quad 0 \leq \frac{n_5}{n_6} \leq \frac{n_7}{n_8} \leq 1$$

we have $h : (n_1, n_2, \ldots, n_8, k) \mapsto (m_1, m_2)$ with $m_1 \leq m_2$ where

$$\frac{m_1}{m_2} \leq \min g\left(\left[\frac{n_1}{n_2}, \frac{n_3}{n_4}\right] \times \left[\frac{n_5}{n_6}, \frac{n_7}{n_8}\right]\right) \leq \frac{m_1}{m_2} + \frac{1}{k}.$$

Let $g = ||f(x) - x||$ and let h be as in the lemma. Note that $g(x) = 0$ if and only if x is a fixed point of f. We shall use only the essential consequences that

- $g(x) > 0$ for all computable x; and
- there exists a (non-computable) x_0 such that $g(x_0) = 0$.

As usual, $\{0,1\}^*$ denotes the set of finite binary sequences and ab is the concatenation of a and b.

Definition 2. *A binary tree is a function $t : \{0,1\}^* \to \{0,1\}$ such that*

$$t(ab) = 0 \quad \textit{for all } b \quad \textit{whenever} \quad t(a) = 0.$$

An infinite branch of a tree t is an infinite binary sequence, on all of which finite initial segments t takes the values 1.

The tree is *computable* whenever the function t is Turing-computable and a *computable branch* is a computable binary sequence which is an infinite branch. Define the Kleene tree as follows. Let

$$t(i_1 \ldots i_n) = \prod_{m=1}^{n} s(i_1 \ldots i_m)$$

where s is a function taking values in $\{0,1\}$. This definition of t ensures that t is in fact a tree and if s is computable, t will be a computable tree. The function s will use h to estimate whether g gets close to zero on a specific square and if g has been bounded away from zero on the square, that branch of the tree will terminate.

Define $s : \{0,1\}^* \to \{0,1\}$ for all sequences $i_1 j_1 \dots i_n j_n$ of even length by

$$s(i_1 j_1 \dots i_n j_n) = \chi_{\{0\}}\left(\frac{m_1}{m_2}\right)$$

where

$$(m_1, m_2) = h\left(i_1 \dots i_n, 2^n, i_1 \dots i_n + 1, 2^n, j_1 \dots j_n, 2^n, j_1 \dots j_n + 1, 2^n, n\right)$$

and binary strings have been interpreted as the natural numbers which they represent. Let s take the value 1 on sequences of odd length.

The tree t defined in this way is obviously computable. It remains to show that t is

- infinite; and
- has no infinite computable branch.

Let x_0 be any point where $g(x_0) = 0$. Then there exist infinite sequences (i_n) and (j_n) such that

$$x_0 \in \left[\frac{i_1 \dots i_n}{2^n}, \frac{j_1 \dots j_n}{2^n}\right] \times \left[\frac{i_1 \dots i_n + 1}{2^n}, \frac{j_1 \dots j_n + 1}{2^n}\right] \quad \text{for all } n$$

and therefore, for all n, $s(i_1 j_1 \dots i_n j_n) = 1$ and so $t(i_1 j_1 \dots i_n j_n) = 1$ which proves the existence of an infinite branch, hence that the tree t is infinite.

Suppose that t had an infinite computable branch. The branch would correspond to a decreasing chain of closed squares, the intersection of which would be non-empty. Let x_1 be a point in the intersection. Since, by construction of the tree, $g(x_1) \leq \frac{1}{n}$ for all n, $g(x_1) = 0$ and hence x_1 would be a fixed point of f. However, by the construction—the branch being computable—the point x_1 would also be computable, contradicting the fact that f has not computable fixed point. Therefore the tree t has no infinite computable branch.

4.2 From Kleene tree to Baigger *f*

Suppose we are given a computable tree t with no infinite computable branch. This tree can be used to construct a sequence of closed intervals with a computable sequence of end-points, covering all the computable real numbers in the unit interval and for which the corresponding open intervals are pair-wise disjoint.

Using the computable function t, one can enumerate all of the maximal finite branches of the tree. Say,

$$b(n) = b_1(n) \dots b_{\lambda(n)}(n)$$

and set

$$J_{n,1} = \left[\frac{b_1(n)\dots b_{\lambda(n)}}{2^{\lambda(n)}}, \frac{b_1(n)\dots b_{\lambda(n)} + \frac{1}{2}}{2^{\lambda(n)}}\right]$$

$$J_{n,m} = \left[\frac{b_1(n)\dots b_{\lambda(n)} + 2^{-m+1}}{2^{\lambda(n)}}, \frac{b_1(n)\dots b_{\lambda(n)} + 2^{-m}}{2^{\lambda(n)}}\right] \quad \text{for } m \geq 2.$$

It remains to show that the union of the intervals $J_{n,m}$ covers *all* the computable points I_c but not all of the unit interval I. It is easy to see that

- for every computable $x \in I_c$ there exists a computable binary sequence (x_n) such that
$$\frac{x_1 \dots x_n}{2^n} \leq x < \frac{x_1 \dots x_n + 1}{2^n} \quad \text{for all } n$$
and since t has no infinite computable branch $t(x_1 \dots x_\ell) = 0$ for some least ℓ, in which case $x \in \cup_m J_{n,m}$ where $b(n) = x_1 \dots x_\ell$;
- if (x_n) is an infinite branch of t then, since it is not computable, for all w we have $x_1 x_2 \dots \neq w1111\dots$ and therefore
$$\lim_n \frac{x_1 \dots x_n + 1}{2^n} \notin \bigcup_m J_{\ell,m}$$
for every ℓ.

The Baigger example f can now be constructed using the intervals $J_{n,m}$ and by that construction one obtains a computable f with no computable fixed point, as required.

5 Conclusion

The existence of the Kleene tree can quite easily be derived from the impossibility of ensuring the existence of a computable fixed point for a computable function (in both Russian and Polish senses), in two dimensions (or higher). The ingenuous constructions of Orevkov and Baigger provide a way of defining a computable function with no computable fixed point from the set of intervals derived from the Kleene tree, in a constructive manner. This correspondence is, perhaps, more attractive for the "working mathematician" than the elegant derivation of the result in reverse mathematics. In one dimension, any computable $f : I \to I$ does have a computable point $x \in I_c$ such that $f(x) = x$, which can be seen by fairly straight-forward reduction *ad absurdum* from the assumption that this is not the case.

References

1. Pour-El, M.B., Richards, J.I.: Computability in analysis and physics. Perspectives in Mathematical Logic. Springer-Verlag, Berlin (1989)
2. Orevkov, V.P.: A constructive map of the square into itself, which moves every constructive point. Dokl. Akad. Nauk SSSR **152** (1963) 55–58
3. Wong, K.C., Richter, M.K.: Non-computability of competitive equilibrium. Economic Theory **14**(1) (1999) 1–27
4. Miller, J.S.: Degrees of unsolvability of continuous functions. J. Symbolic Logic **69**(2) (2004) 555–584
5. Baigger, G.: Die Nichtkonstruktivität des Brouwerschen Fixpunktsatzes. Arch. Math. Logik Grundlag. **25**(3-4) (1985) 183–188
6. Shioji, N., Tanaka, K.: Fixed point theory in weak second-order arithmetic. Ann. Pure Appl. Logic **47**(2) (1990) 167–188
7. Kleene, S.C.: Recursive functions and intuitionistic mathematics, Providence, R. I., Amer. Math. Soc. (1952) 679–685
8. Simpson, S.G.: Which set existence axioms are needed to prove the cauchy/peano theorem for ordinary differential equations? The Journal of Symbolic Logic **49** (1984) 783–802
9. Simpson, S.G.: Subsystems of second order arithmetic. Perspectives in Mathematical Logic. Springer-Verlag, Berlin (1999)

Appendix A: details of the construction in Section 3

The constructions should guarantee that at each stage, the function f_n moves every point of

$$D_n = \left(C_n^{\blacksquare 2^{-n}} \setminus C_n^{\blacksquare 2^{-n}\cdot\frac{5}{4}}\right)^\circ$$

in the direction of t_n by an amount proportional to its distance to $C_n^{\square 2^{-n}}$. The construction of f_1 with this property is trivial. We proceed to construct f_{n+1} from f_n.

(i) Extend and modify f_n to $C_{n+1}^{\blacksquare 2^{-n}}$ so that every point x of

$$\left(C_{n+1}^{\blacksquare 2^{-n}} \setminus C_{n+1}^{\blacksquare 2^{-n}\cdot\frac{5}{4}}\right)^\circ$$

is moved in the direction of t_n by an amount proportional to $d\left(x, C_{n+1}^{\square 2^{-n}}\right)$.

(ii) Modify the resulting function so that each point in

$$C_{n+1}^{\blacksquare 2^{-n}} \setminus C_{n+1}^{\blacksquare 2^{-n}\cdot\frac{9}{8}}$$

is mapped a non-negative amount proportional to its distance to $C_{n+1}^{\square 2^{-(n+1)}}$ in the direction of t_n.

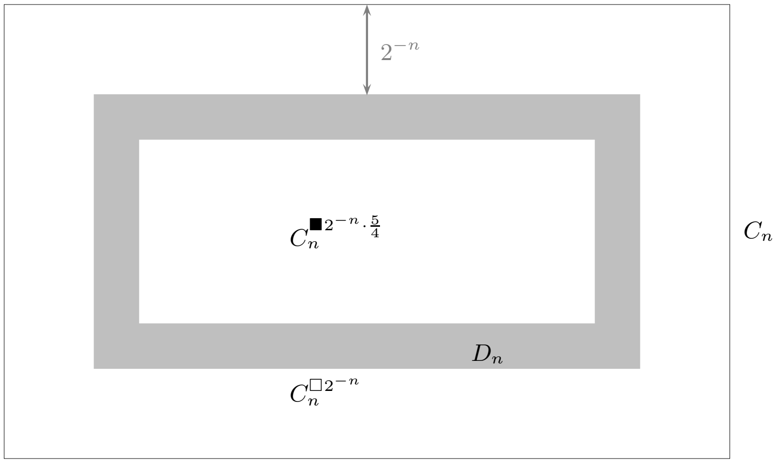

Fig. 3. Sets used in the construction

(iii) By rotation of the direction of the mapping, extend the function to $C_{n+1}^{\blacksquare 2^{-(n+1)}}$ such that every point x of

$$D_{n+1} = \left(C_{n+1}^{\blacksquare 2^{-(n+1)}} \setminus C_{n+1}^{\blacksquare 2^{-(n+1)}\cdot\frac{5}{4}} \right)^{\circ}$$

is mapped in the direction of t_{n+1} by an amount proportional to $d\left(x, C_{n+1}^{\square 2^{-(n+1)}}\right)$.

The final step is the only one in which we use the fact that we are working in two dimensions as this step requires the continuous (computable) rotation of a vector in the direction of t_n to a vector in the direction of t_{n+1}.

A construction is given explicitly in [5] but it should be clear from the preceding that it can be done in many different ways. The important part of the proof is that the construction is, at each stage, extended at the boundary to "look right" from the outside. This ensures that, eventually every point is in fact moved towards one of a sequence of points that converge to the non-computable fixed point (v, v) on the diagonal. The Baigger construction is a somewhat delicate construction of a function that is in fact computable but that—somehow—mimics a simple mapping of every point in I^2 in the direction of (v, v).